Queue management in packet switched networks

1. Field of invention

The invention relates to media access control (MAC) in packet switched networks. It is particularly directed to the management of the transfer of packets between a central node and one of several multipoint nodes over a shared medium.

2. Background art

Media access control of a point-to-multipoint link in a packet switched network is a mechanism for determining which of a plurality of nodes at the multipoint end of the link is allowed to send a packet over the shared medium to a single central node located at the other end of the link. In a conventional media access control mechanism utilised in two-way TDMA transfer links, the central node appends permit information, commonly called a permit, to every packet that it sends. The permit is addressed to one of the multipoint nodes and specifies which time slot in the future packet stream can be utilised by the addressed multipoint node to send a packet.

Permits are allocated in response to requests. The multipoint nodes send requests, typically in the form of queue size information, when packets are present at the node awaiting transfer. The central node sends corresponding permits and will also manage the allocation of priorities when the requests for permits arrive more frequently than permits can be sent. To this end, each multipoint node has a queue that stores packets until the node is permitted to

pass them on.

Each multipoint node records the number of queued packets awaiting transmission so that a corresponding permit request may be sent to the central node. The central node also holds a record of the queue lengths at each multipoint node. On the basis of this queue record, the central node issues permits to those multipoint nodes requiring channel capacity.

The queue record in the central node is updated by queue information in the form of a permit request received from the multipoint node. This request represents the current queue size at the multipoint node and is used to overwrite the recorded queue size at the central node. The recorded queue size at the central node is also decremented each time a permit to send a packet is transmitted to the corresponding multipoint node.

The transfer of information between a central node and a multipoint node involves a delay, which depends on the available bandwidth and length of the transfer medium. A delay also occurs at the multipoint node when processing permits received from the central node. Hence there exists a window during which a multipoint node may transmit a permit request and the central node may transmit permits to the same multipoint node on the basis of the queue record before receiving the permit request. When the permit request reaches the central node, the queue record will be overwritten by an erroneous queue size value. While this error will not cause the loss of packets, it can lead to more permits being sent than are actually required. It is possible that the discrepancy between the actual and recorded queue size will be mitigated to some extent by further packets arriving at the multipoint node, however, on balance there is a high probability that the central node will issue permits that

cannot be used. Channel capacity will thus be wasted.

SUMMARY OF INVENTION

It is an object of the present invention to provide a method and arrangement which alleviates the problems of the prior art.

It is a further object of the present invention to provide a method and arrangement which substantially improves the utilisation of channel capacity in a multipoint-to-point link.

This object is achieved with a method whereby a traffic receiving node, which holds a queue record of the number of packets awaiting transfer at a traffic sending node, stores the number of permits, or the equivalent number of packet transfers enabled by the permits, for a predetermined delay after transmission of each permit, and modifies the value of any incoming requests for permits by the number of permits stored. The delay preferably corresponds to the round trip delay between the two nodes in question. This is the sum of the permit delay and the report delay, i.e. the total time between a permit being sent by the traffic receiving node and the corresponding (permitted) traffic packet being sent by the traffic sending node. This delay also includes the processing delay required by the traffic sending node to recognise, or process, the permit.

The invention further resides in a node in a communication network that receives traffic from at least one further node over a shared medium. The node is arranged to store the number of permits sent to the further node, or the equivalent number of packet transfers enabled with these permits, for a predetermined delay after transmission of each permit, and to modify any

incoming queue size report from the same further node by the value of stored permits.

There is further proposed a communication network including a central node and several multipoint nodes, the multipoint nodes being connected to said central node by a shared medium. The central node holds a queue size record for each multipoint node and includes means for storing the number of permits sent to respective multipoint nodes , or the equivalent number of packet transfers enabled, for a predetermined delay and modifying any incoming queue size records corresponding to the multipoint node in question by the value of stored permits.

Storing the number of permits sent in the preceding round trip delay effectively establishes the number of permits that can not yet have been processed and responded to by the corresponding multipoint node. Any queue size record, or request for permits, received from the corresponding multipoint node within this period will have been issued without knowledge of the permits. By modifying this record value using the number of permits sent, or the equivalent number of packets enabled if one permit allows more than one packet to be transmitted, the reported queue size will be corrected and the correct value used to overwrite the queue record in the central node.

By means of the method and the node and network arrangements according to the invention, the queue size record in the traffic receiving node will be constantly adjusted to reflect the actual number of permits required. The conventional overestimation of queue size and the resulting allocation of unused time slots will thus be eliminated. Consequently, the transfer capacity and efficiency of a network will be improved.

BRIEF DESCRIPTION OF THE DRAWINGS

Further objects and advantages of the present invention will become apparent from the following description of the preferred embodiments that are given by way of example with reference to the accompanying drawings, in which:

Fig. 1 schematically depicts a multipoint-to-point link in a TDMA packet switched network.

Figs. 2a to 2k schematically depict the process conventionally utilised for recording the size of a packet queue located at multipoint node in a central node in the network of Fig. 1 , and

Figs. 3a to 3l schematically shows the process for adjusting the queue record at a central node in a multipoint-to-point network of Fig. 1 in accordance with the invention.

DETAILED DESCRIPTION OF THE DRAWINGS

In the figures, like reference numerals are used for similar or identical components.

A typical multipoint-to-point link in a TDMA packet switched communication network is illustrated in Fig. 1. Several multipoint nodes 20, three of which are shown in the figure, communicate via a single, shared bi-directional channel

with a central node 10. The channel 100 may be a fixed electrical or electro-optical link or an air interface.

A media access control (MAC) mechanism is used by the central node 10 to determine which of the multipoint nodes may use which time slots on the shared channel 100 to transmit packets. It is assumed here that a packet is the amount of information that may be transmitted over the channel 100 in one time slot. Depending on the services utilising this channel, the packet may contain one or several frames, or parts of a frame.

A conventional media access control procedure is illustrated in Figs. 2a to 2k. Figs. 2a to 2k show a greatly simplified view of the flow of frames or packets between a central node 10 and one multipoint node 20 in a link as shown in Fig. 1. Each frame is sent in one time slot 30. In each of the Figs. 2a to 2k the downstream link is shown at the top and the upstream link at the bottom. The dashed lines represent the interfaces between the channel 100 and each of the nodes 10, 20.

It is assumed in the illustrated link that the transmission delay or time of flight between the two nodes is equal to three slot periods. The processing time in the multipoint node is assumed to be equal to a single slot period 30. This is represented by the time slot 31 at the multipoint node 20. Each consecutive figure represents the link at intervals of one time slot.

Fig. 2a shows the starting state wherein the multipoint node has a queue length of 6 packets waiting to be sent, and the central node 10 holds a record 11 of this queue length. The central node 10 has just sent a permit to send a request RP to the multipoint node 20. This permit to send a request RP is an invitation

to the multipoint node 20 to indicate via a request, Rn, for traffic permits, TP, the number n of permits that are required to send all packets in the queue. The request for permits Rn is therefore effectively a queue size report. In the example illustrated, one permit represents one time slot during which a packet can be sent. However it will be understood that a permit could represent an invitation to send more than one packet, for example in consecutive time slots. Moving on to Fig. 2b, it can be seen that the permit to send a request RP has advanced by one time slot delay across the link, and that the central node 10 has followed this by sending a first traffic permit TP. The issue of this traffic permit TP by the central node 10 has caused the recorded queue length 11 to be decremented by one. However, the actual queue in the multipoint node is still 6, because no packets have yet been sent.

In Fig. 2c a further traffic permit TP has been issued by the central node 10, and the recorded queue length amended to 4. The permit to send a request RP is still in transmission. In Fig. 2d the permit to send a request RP has reached the multipoint node 20 and is being processed. As mentioned above, this will take one slot period as represented by the slot 31. In the meantime, the central node 10 has issued a third traffic permit TP and the recorded queue size 11 has been reduced to 3. In Fig. 2e the permit to send a request RP has been processed by the multipoint node 20 and has resulted in a request for six permits, R6, being sent. The first traffic permit TP has also reached the multipoint node 20, 31 and is ready for processing. At the same time, the central node has issued a fourth traffic permit TP and has modified the recorded queue size to 2.

In Fig. 2f, the multipoint node 20 has processed the first traffic permit TP and has sent the first traffic packet in response. The queue at the multipoint node

20 now contains 5 packets. The central node has issued a fifth traffic permit TP, and the queue record 11 has been modified to 1. The queue record is finally reduced to zero in Fig. 2g as a final sixth traffic permit TP is sent. The multipoint node 20 has also responded to the second traffic permit TP by sending a second packet so that the actual queue size at the node 20 is reduced to 4. In Fig. 2h, the central node has received the request for permits R6 which overwrites the queue record 11. The latter is therefore updated to 6. In the meantime the multipoint node 20 continues to send packets in response to the permits. This continues through Figs. 2i to 2k until all the packets at the multipoint node 20 have been sent in response to the six traffic permits TP. The final queue at the multipoint node 20 is thus zero. However, the recorded queue size 11 at central node 10 is six.

In this particular example, the error in the queue size recording at the central node 10 results from a request for traffic permits Rn being transmitted on the downlink and permits being transmitted on the uplink during the same period. Effectively both nodes are issuing information in ignorance of information that is in transmission.

Figs. 3a to 3l show the procedure for removing this source of error in accordance with the present invention. In this arrangement, the central node 10 includes a permit memory 12, which may be any temporary storage medium such as a register or a counter or equivalent, for recording the number of traffic permits TP sent for a predetermined period after their emission. This memory 12 effectively stores a history of the traffic permits TP issued in the preceding round trip delay to the multipoint node 20. In the present embodiment, this means that the register 12 is adapted to hold the number of permits sent within the last preceding 7 slot periods. These 7 slot periods correspond to the round

trip delay between the central node 10 and a the multipoint node 20. Specifically, the delay is equal to the time required for a traffic permit TP to reach, and be processed by, the multipoint node 31 plus the transmission delay for a traffic packet T sent by the multipoint node 20 to reach the central node 10.

As for the procedure of Figs. 2a to 2k it is assumed that the delay for processing the request for permits Rn, i.e. for updating the recorded queue size, in the central node is negligible. The procedure according to the invention will now be described with reference to the Figs. 3a to 3k. These show the same series of events illustrated in Figs. 2a to 2k for the arrangement according to the invention. Fig. 3l shows the link 3 time slot delays after Fig. 3k.

In Fig. 3a a queue of 6 packets are waiting at the multipoint node 20. The central node 10 holds a record 11 of this queue size, and has also just issued a permit to send requests RP. The permit memory 12 is empty because no traffic permits TP have been sent during the preceding 7 slot periods.

In Fig. 3b a first traffic permit TP is transmitted to follow the permit to send requests RP. This event causes the queue record 11 to be decremented by one. The permit memory 12 is also incremented and holds the value 1.

During the following three time slots 30 (Figs. 3c to 3e) a further three traffic permits TP are issued. The resultant queue record 11 is therefore 2 and the value in the permit memory 12 is 4. In the meantime, the multipoint node 20 has issued a reply to the RP in the form of a request for six permits R6. While this reply R6 is travelling towards the central node 10, two further traffic

permits TP are issued (Fig. 3f and 3g) reducing the queue record 11 to zero and bringing the value in the permit memory to 6. At the end of this time slot, the request for six permits R6 has reached the central node 10, this is illustrated in Fig. 3h which shows the situation in the subsequent time slot. It will be remembered that the processing time in the central node is considered to be negligible. However, before the request for permits R6 is used to overwrite the queue record, its value is first adjusted by the number of permits stored in the permit memory 12. Six permits are stored in the memory since the first permit was transmitted 7 time slots previously. The stored permits are subtracted from the request value R6 resulting in a request value of zero. This value is then used to update, or overwrite the recorded queue value 11. Thus the recorded queue value 11 effectively remains unchanged. In the next slot period as shown in Fig. 3i the round-trip delay of 7 time slots has expired for the first traffic permit TP. The permit memory is therefore decremented by one. In Figs. 3j and 3k the permit memory 12 is decremented further as two more permit delays expire. Finally Fig. 3l shows the state three time slot delays subsequently when the last permit delay has expired and the permit memory 12 is finally set to zero.

It will be understood that the time period during which the permit memory 12 will actually store the records of permits will exceed 7 time slots, because there will inevitably be some finite processing delay associated with the central nod 10.

As a result of the adjustment of the queue size report (permit request) the recorded queue size 11 in the central node 10 will always provide an accurate representation of the number of packets awaiting transmission at the multipoint node 20.

For the multipoint-to-point link shown in Fig. 1, the central node 10 will hold a separate record 11 of the queue size at each of the multipoint nodes 20. Furthermore, the central node will store the number of traffic permits issued during the preceding round trip delay for each individual multipoint node 20. It will be understood that different multipoint nodes may be associated with different round trip delays. Each permit memory 12, or that portion of memory assigned to permits, allocated to a specific multipoint node 20 will thus be adapted to store the permit transmission history for the corresponding delay.

In order to eliminate the risk of the central node 10 underestimating the number of packets waiting at a multipoint node 20, the period during which the permits sent are recorded should be shorter than the round trip delay in terms of frames between the two nodes plus one time slot delay. This prevents any request for permits Rn reporting newly arrived packets from being reduced or decremented by the number of stored permits erroneously. In terms of the simplified model shown in Figs. 2 and 3, the delay should be less than 8 time slots but at least 7 time slots.

It will be understood that each multipoint node 20 may hold more than one queue, such that different packets awaiting transmission are assigned to different queues and treated with different priority. The central node 10 will also hold these different queue values in the queue record 11 allocated to the multipoint node. In this case, the number of permits issued for packets of different queues will be stored separately and this record used to adjust the corresponding queue size value in the queue record.